

SunBeam Institute of Information Technology

Java Assignment 7

Based on Exception Handling

Q1) Define a new exception, called `ExceptionLineTooLong`, that prints out the error message "The strings is too long". Write a program that reads a `String` from user and calculates its length. and throws an exception of type `ExceptionLineTooLong` in the case where a string of length is more than 80 characters.

Q2) Build a new `Circle` class with the following basic features:

- Attributes:
 - Center point - Build instance variables for the circle's center point (myX and myY represented as double values).
 - Diameter - Build an instance variable for the circle's diameter (myDiameter also represented as a double value).
- Behaviors
 - Default constructor - Build a default constructor that initializes the circle's center point to (0, 0) and its diameter to 100.
 - Accessor methods - Build accessor methods for the two center coordinates and the diameter.
- Invariant
 - The circle's diameter should always be non-negative. maintain the integrity of each circle object by ensuring that the class invariant (that the diameter should be non-negative) is true at all times.

If the diameter is negative then throw user defined exception.

OPTIONAL Assignment

Solve Vehicle equality assignment

Problem : Create a class Vehicle , under the package : com.app.vehicles , to represent any vehicle.

Supply tight encapsulation

state(data members) : chasisNo : int , color : String , price : double

chasisNo MUST be unique (non duplicate) for any vehicle (primary Key : unique ID)

Get complete state : toString

Hint : override equals method inherited from Object class : to replace ref equality by content equality(based upon unique ID : PK)

Create a class TestVehicle under the package "tester"

Accept details of 2 vehicles from user .

Display "SAME" or "DIFFERENT" , in case of same or different chasis nos.

Hint : invoke equals

Objective : Use exception handling , string handling & Date in vehicle showroom based scenario

Vehicle's state(attributes) --chasisNo : String , color : String ,category:string, price : double , manufactureDate(Date)

Add validation rules for the inputs :

chasisNo id unique

category : Petrol , Diesel, EV , Hybrid , CNG

manufactureDate : in this financial year (1st Apr 2021 to 31Mar 2022)

Create a main class as Showroom. By giving Options

1. Add Vehicle

Apply all validation rules . In case of success , add the vehicle in the showroom Otherwise raise custom exception , alert user using exc handling.

2. Display All Vehicles

3. Exit